

GitHub 習熟度チェックリストの作成と システム開発 PBL における習熟度および活動の評価

伊藤 恵^{*1}, 松原 克弥^{*1}

^{*1} 公立はこだて未来大学

Creating a Checklist to Evaluate Learning Level of GitHub and Evaluating System Development PBL

Kei Ito^{*1}, Katsuya Matsubara^{*1}

^{*1} Future University Hakodate

In system development PBLs, many student teams use several standard tools, such as GitHub. By usage history data of such tools, we aim to evaluate objectively and quantitatively students' activities in PBL. However, evaluation using usage history data of such tools is greatly affected by proficiency level of students. So, it is also needed to evaluate of proficiency level of students. For that reason, we create a checklist for proficiency level of GitHub and try to evaluate proficiency level by the checklist in a certain GitHub workshop and an on-campus Hackathon event. In this article, we report the checklist and the evaluation results by the checklist.

キーワード: 習熟度チェックリスト, システム開発 PBL, GitHub, 習熟度評価

1. はじめに

さまざまな教育分野において実践的な教育方法の一つとして Project Based Learning(以下 PBL)が行われており, 情報系の学部や学科においても, 実践的な IT 教育の一つとしてシステム開発 PBL が広く実施されるようになってきている⁽¹⁾⁽²⁾. このようなシステム開発 PBL における成績評価は発表会や最終成果物に対する評価が中心である. しかし, 学習対象には発表や最終成果物だけでなく, そこに至るまでのプロセスも含まれていることが多い. システム開発 PBL において開発プロセスの評価方法は確立されているとは言えない⁽³⁾.

システム開発 PBL では GitHub などの定番ツールがよく使われる. GitHub¹はシステム開発等で使われるバージョン管理ツールの一つ Git を基にした Web 上のサービスであり, 開発プロセスに紐づけられる利用履歴情報が相応に残される. 我々はその情報に着目

し, GitHub を使用するシステム開発 PBL を対象に, 利用履歴情報を用いた PBL プロセス評価手法の確立を目指している⁽⁴⁾. しかし, Git/GitHub はまあまあ使えるレベル, かなり使いこなしているレベルなど, 使用者の習熟度合の幅が広く, システム開発 PBL においてもチーム間やチーム内で習熟度合の差が大きくなることも多く, それが利用履歴を用いたプロセス評価に大きく影響してしまう.

そこで我々は, システム開発 PBL のプロセス評価に活用することを前提として GitHub の習熟度チェックリストを作成し, 実際にこれを用いた習熟度とプロセスの評価実験を行うこととした.

2. GitHub 習熟度チェックリストの作成

2.1 Git/GitHub とは

GitHub とは, ソフトウェア開発その他の活動の際にバージョン管理のために用いられる Git を活用し,

¹ <https://github.com>

インターネットを通じた共同開発や開発したソフトウェアの公開などを行うことができる Web 上のサービスである。

GitHub の習熟度は、Git の基本的なコマンドが使えるレベル、円滑な共同開発のためにブランチという仕組みを適切に使えるレベル、Git Flow や GitHub Flow などの共同開発のワークフローを活用できるレベルなど様々あり、習熟度の表現や計測方法は確立されていない。

2.2 習熟項目のリストアップ

習熟度チェックリストの作成に先立って、GitHub に関する参考書籍⁽⁵⁾や関連する Web ページ⁽⁶⁾、GitHub にある程度習熟している教員や学生複数の知見を参考に、GitHub の習熟度合の参考となる項目をリストアップし、大まかに基本操作に関するもの、円滑な共同開発のために重要となるブランチに関するものに分類した。その他に共同開発のワークフローに関するものを検討したが、ワークフローについては個人単位での評価が難しいことなどから、今回は項目化の対象から除外した。各分類の習熟項目は以下の通りである。

● 基本操作

コミットができる
コミット間の diff がみれる
リモートのリポジトリにプッシュができる
リモートのリポジトリから pull ができる
リモートのリポジトリのクローンができる
git commit --amend ときいて使い方がわかる
.gitignore が何のためにあるファイルか知っている
コミットをどの程度の粒度にすべきか分かる
コミットメッセージにはどんな内容を書くべきか分かる

● ブランチ

ブランチの作成ができる
ブランチの削除ができる
あるブランチとあるブランチ 2 の差分がすぐに出せることを知っている
あるブランチ A とあるブランチ B を比較して、A のみに加えられた差分をすぐに出せることを知っている

merge ができる
merge に失敗した後、merge 前に戻るやりかたを知っている
rebase ができる
rebase に失敗した後、merge 前に戻るやり方を知っている
merge 中のコンフリクトを解消できる
どんな名前がブランチ名として適しているか分かる
どのくらいの頻度(単位)でマージするのがよいか分かる

2.3 習熟度判定基準の作成

この時点では、GitHub にある程度習熟している複数の教員と学生の知見を踏まえて、GitHub の習熟度を基本操作もままならない「入門者」、基本操作は一通りできる「初級者」、ブランチについてもある程度精通した「中級者」、ブランチについてはほぼすべて把握している「上級者」の 4 段階とし、各項目の習熟度合を計測することで、対象者の習熟度を算定できるものとした。

3. 学内ハッカソンと GitHub 勉強会

本研究で目指すのはシステム開発 PBL におけるプロセス評価であるが、評価実験として 2019 年 2 月 16、17 日に著者ら所属大学で行われる教育目的の学内ハッカソン P2HACKS⁽⁷⁾と、それに先立って 2018 年 12 月に行われた Git/GitHub 勉強会を対象とした評価実験を行うこととした。

ハッカソンはシステム開発 PBL に比べて極端に期間が短いなどの違いはあるが、チームで開発を行う点ではシステム開発 PBL と同様であること、今回対象としたハッカソンは学部 1,2 年生のスキルアップを主眼とした育成型ハッカソンであったことから評価実験の対象として十分有効であると判断した。

また、短期集中の活動であるため、収集するデータへの他活動からの影響がほとんどないことや、事前に Git/GitHub に関する勉強会が開催されることも実験対象として有用であった。

このハッカソンには、1,2 年生のみ 4~5 名からなる

チームが 10 チーム計 48 名参加し、そのうち Git/GitHub 勉強会に参加したのは 20 名であった。勉強会にメンバ全員が参加したチームもあったが、メンバが 1 人も参加しないチームもあった。Git や GitHub は著者ら所属大学の授業等で明に扱っていないが、学内の何らかの活動で利用している学生もいる可能性があることから、勉強会に参加した学生は Git 等の使用経験がほとんどなく、ハッカソンには参加するが勉強会には参加しない学生は使用経験が相応にある学生が多いのではないかと考えられる。

4. チェックリストを用いた評価実験

4.1 習熟度評価：勉強会前

ハッカソン参加学生全員を対象として勉強会前の時点で、2.2 節で挙げた習熟項目のうち、基本操作に関する 9 項目とブランチに関する 11 項目の計 20 項目に対して、「何のことか分からない」「コマンドは分かる/使ったことはある」「分かる/使っている」「使いこなしている/十分理解している」の 4 択でアンケート調査を行った。アンケート回答者は 20 名で、そのうち勉強会参加者は 9 名、残り 11 名は勉強会不参加者であった。

アンケート結果を「何のことか分からない」を 0 点、「コマンドは分かる/使ったことはある」を 1 点、「分かる/使っている」を 2 点、「使いこなしている/十分理解している」を 3 点として集計したところ、回答した学生別の全項目および分類別の平均は表 1 のようになった。全項目平均の降順に並べており、各列の意味は、学生の通し番号、全項目の平均値、基本操作に関する項目の平均値、ブランチに関する項目の平均値、勉強会参加の有無となっている。全体の傾向としては勉強会不参加の方が数値が高く、勉強会参加者の方が低いと言えるが、勉強会不参加者の中でも数値が低い者もあり、逆に勉強会参加者の中でもある程度は数値が高いものもいることが分かる。

また、2.3 節で述べたように初級者は基本操作が分かり、中級者になるとブランチも分かってくると想定しており、例えば、表の学生 1 は基本操作の数値もブランチの数値も高いので上級者、学生 2,3 は基本操作の数値は高いがブランチの数値やそれよりもやや低い

ため中級者などと判断できると考えた。しかし、学生 4 以降もすべて見ていくと基本操作とブランチの数値が逆転している学生がいるほか、どちらの数値も揃って下がっていく傾向が見られるため、習熟の流れとして基本操作の数値が十分上がってからブランチの数値が上がるとは言えないことが分かる。

表 1 アンケート結果(勉強会前学生別)

	全体	基本操作	ブランチ	勉強会参加
学生 1	3.00	3.00	3.00	0
学生 2	2.20	2.56	1.91	0
学生 3	1.90	2.22	1.64	1
学生 4	1.55	0.67	2.27	0
学生 5	1.35	1.56	1.18	1
学生 6	1.15	1.44	0.91	0
学生 7	1.15	1.33	1.00	0
学生 8	1.00	1.44	0.64	0
学生 9	0.75	0.78	0.73	0
学生 10	0.40	0.67	0.18	1
学生 11	0.40	0.56	0.27	0
学生 12	0.40	0.56	0.27	1
学生 13	0.20	0.11	0.27	0
学生 14	0.00	0.00	0.00	1
学生 15	0.00	0.00	0.00	1
学生 16	0.00	0.00	0.00	1
学生 17	0.00	0.00	0.00	0
学生 18	0.00	0.00	0.00	0
学生 19	0.00	0.00	0.00	1
学生 20	0.00	0.00	0.00	1

項目間の傾向や勉強会参加者/不参加者の傾向の違いを見るために、同じく勉強会前のアンケート結果から項目別の回答者平均値を出したものが表 4 である。回答者全員の平均では「コミットができる」「リモートのリポジトリのクローンができる」などの Git の使い始めに使用する操作が 0.89 点と他の項目よりやや高い結果であり、分類別では基本操作に関する項目の数値よりもブランチに関する項目の数値の方がやや低い傾向であった。Git/GitHub 勉強会に参加した学生と参加していない学生に分けて集計した平均値が表 4 の 3,4 列目である。見て明らかな通り、すべての項目において勉強会に参加していない学生の数値が高く、勉強

会に参加しなかった学生は、この時点である程度 Git/GitHub の使用経験があったのではないかと推測される。また、相対的に習熟度が高いと推測される勉強会不参加者でも数値が低い基本操作に関する項目がある一方で、習熟度が低いと推測される勉強会参加者でも数値が高いブランチに関する項目があることが分かる。

これらの結果を踏まえ、2.2 節で挙げた項目の分類のままではなく、分類も参考にしつつ各項目の習熟し易さや習熟に掛かる時間を考慮して習熟度判定基準を再設計する必要があると言える。

4.2 習熟度評価：ハッカソン前

Git/GitHub 勉強会開催後、ハッカソン実施前にも同様にアンケート調査を行った。回答者のうち、勉強会参加者のみの回答者別平均値をまとめたものが表 2 である。勉強会後に数値が上がっている箇所にはセルの背景色を薄灰色にした。この結果から該当者のほとんどは勉強会前後で数値が向上しているが、勉強会で比較的容易に上がると考えられる基本操作の数値が全く上がっていない学生や、むしろブランチの数値の方が上がっている学生、また、数値が下がっている学生がいることも見て取れる。

表 2 アンケート結果(勉強会前後学生別)

	勉強会前			勉強会後		
	全体	基本操作	ブランチ	全体	基本操作	ブランチ
学生 3	1.90	2.22	1.64	2.05	2.33	1.82
学生 5	1.35	1.56	1.18	1.75	1.89	1.64
学生 10	0.40	0.67	0.18	0.50	0.67	0.36
学生 12	0.40	0.56	0.27	0.35	0.56	0.18
学生 14	0.00	0.00	0.00	0.65	0.89	0.45
学生 15	0.00	0.00	0.00	0.25	0.00	0.45
学生 16	0.00	0.00	0.00	0.60	0.67	0.55
学生 19	0.00	0.00	0.00	1.00	1.00	1.00
学生 20	0.00	0.00	0.00	0.85	1.44	0.36

4.3 習熟度判定基準の調整とハッカソン時の評価

4.1 節および 4.2 節の結果を踏まえ、習熟度判定基準の評価方法について再検討を行った。具体的には基

本操作とブランチのそれぞれの分類の中で、比較的早く習得できると思われる項目とそうでない項目を再分類し、基本操作 1、基本操作 2、ブランチ 1、ブランチ 2 の 4 分類に分け、新しい分類に基づいて勉強会前後のアンケート結果を再度学生別に集計した。その結果が表 3 である。この結果から各学生の 4 分類それぞれの平均数値が 1 未満のもの、1 以上 2 未満のもの、2 以上のものがそれぞれ何個あったかで判定基準を再検討した。具体的には、4 分類すべて 1 未満ならば「入門者」、4 分類中半分以上が 1 未満ならば「初級者」、半分以上が 1 以上ならば「中級者」、半分以上が 2 以上ならば「上級者」などといったものである。表 3 では学生 1,2 が上級者、学生 3,6,7,8 が中級者、学生 4,5,9,11,12 が初級者、それ以外が入門者となる。ただし、まだデータ数も十分ではないため網羅的な基準として整備しておらず、習熟度判定の仮基準としている。

表 3 再分類後の結果(勉強会前学生別)

	全体	基本操作 1	基本操作 2	ブランチ 1	ブランチ 2
学生 1	3.00	3.00	3.00	3.00	3.00
学生 2	2.20	3.00	2.00	2.40	1.83
学生 3	1.90	2.60	1.75	2.00	1.17
学生 4	1.55	0.60	0.75	2.00	2.50
学生 5	1.35	2.60	0.25	2.20	0.17
学生 6	1.15	1.80	1.00	1.00	0.67
学生 7	1.15	1.60	1.00	1.00	1.00
学生 8	1.00	1.80	1.00	1.20	0.17
学生 9	0.75	1.00	0.50	0.80	0.67
学生 10	0.40	0.80	0.50	0.40	0.00
学生 11	0.40	1.00	0.00	0.60	0.00
学生 12	0.40	1.00	0.00	0.60	0.00
学生 13	0.20	0.20	0.00	0.60	0.00
学生 14	0.00	0.00	0.00	0.00	0.00
学生 15	0.00	0.00	0.00	0.00	0.00
学生 16	0.00	0.00	0.00	0.00	0.00
学生 17	0.00	0.00	0.00	0.00	0.00
学生 18	0.00	0.00	0.00	0.00	0.00
学生 19	0.00	0.00	0.00	0.00	0.00
学生 20	0.00	0.00	0.00	0.00	0.00

また、ハッカソン終了後にも同様に習熟度評価のためのアンケート調査を行う予定であるが、本稿の執筆時点ではハッカソン実施前のため、結果を示せない。

さらに、ハッカソン時には参加チームごとの GitHub 利用履歴情報を収集し、先に行っている PBL チーム活動評価⁽⁴⁾を踏まえ、ハッカソン参加各チームの評価を行う予定である。

5. おわりに

システム開発 PBL の定量的なプロセス評価方法の一つとして、システム開発で標準的に使われる GitHub の利用履歴情報を用いた評価手法の作成を進めている。その前段階として PBL 参加学生および参加チームの GitHub 習熟度を測定するため、GitHub 習熟度チェックリストを作成した。その評価実験として、教育目的の学内ハッカソンに向けて実施された GitHub 勉強会の前後でアンケートによる主観評価からの習熟度測定を行った。習熟度判定の基準として不十分な点もあり、評価実験を継続しながら判定基準の改善を図っている。

参 考 文 献

- (1) enPiT 事務局: “高度 IT 人材を育成する産学協働の実践教育ネットワーク”, <http://www.enpit.jp>, (2017)
- (2) 福田晃, 鶴林尚靖ほか: “情報工学系大学教員のための PBL 実践ガイド”, 九州大学大学院システム情報科学府 情報知能工学専攻社会情報システム工学コース (2012)
- (3) 日戸直紘, 伊藤恵, 大場みち子: 能力成熟度モデル統合に基づいた PBL における定量的学習評価手法の提案, 日本ソフトウェア科学会第 34 回大会 (2017)
- (4) 松原克弥, 伊藤恵, 木塚あゆみ: “コード更新履歴の可視化による分散 PBL でのチーム活動評価の試み”, 第 4 回実践的 IT 教育シンポジウム(rePiT2018), pp.49-55 (2018)
- (5) 大塚弘記: “GitHub 実践入門”, 技術評論社 (2014)
- (6) @kyanro@github: “git 理解度チェックリスト”, <https://qiita.com/kyanro@github/items/3f0d4303882e668e1e85> (2019 年 2 月 6 日アクセス)
- (7) P2HACKS 実行委員会: “P2HACKS – プレ PBL としての育成型ハッカソン”, <https://p2hacks.c.fun.ac.jp/> (2019 年 2 月 6 日アクセス)

表 4 アンケート結果（勉強会前項目別）

	回答者 平均	勉強会 参加者	勉強会不 参加者
コミットができる	1.15	0.89	1.36
コミット間の diff がみれる	0.55	0.22	0.82
リモートのリポジトリにプッシュができる	1.10	0.89	1.27
リモートのリポジトリから pull ができる	1.05	0.67	1.36
リモートのリポジトリのクローンができる	1.10	0.89	1.27
git commit --amend ときいて使い方がわかる	0.50	0.22	0.73
.gitignore が何のためにあるファイルか知っている	0.65	0.44	0.82
コミットをどの程度の粒度にすべきか分かる	0.65	0.22	1.00
コミットメッセージにはどんな内容を書くべきか分かる	0.85	0.56	1.09
ブランチの作成ができる	1.05	0.89	1.18
ブランチの削除ができる	0.85	0.44	1.18
あるブランチとあるブランチ 2 の差分がすぐに出せることを知っている	0.60	0.22	0.91
あるブランチ A とあるブランチ B を比較して、A のみに加えられた差分をすぐに出せることを知っている	0.50	0.11	0.82
merge ができる	1.05	0.67	1.36
merge に失敗した後、merge 前に戻るやりかたを知っている	0.50	0.11	0.82
rebase ができる	0.75	0.44	1.00
rebase に失敗した後、merge 前に戻るやり方を知っている	0.55	0.22	0.82
merge 中のコンフリクトを解消できる	0.60	0.11	1.00
どんな名前がブランチ名として適しているか分かる	0.75	0.44	1.00
どのくらいの頻度(単位)でマージするのがよいか分かる	0.65	0.33	0.91